Announcing
JCC LML *for Oracle Sources*

Jeffrey S. Jalbert
Cheryl P. Jalbert ♦ Thomas H. Musson ♦
Keith W. Hare
Jeffrey S. Haidet

JCC Consulting, Inc.

Copyright 2010, JCC Consulting, Inc. All rights reserved

1

JCC LML for Oracle Topics

- What is the JCC LogMiner Loader?

- What are the differences in Rdb and Oracle 10, 11, etc. that caused us to wait all these years to add Oracle as a source?

- What architectures do we envision using the new product?

- What are the architectural limits to using both products together?

- How is the new product constructed? What do you need to know to use it?

- What are the results from early testing?

- What is the expected delivery schedule?

< Pause >

- How many of you are working where the JCC LogMiner Loader is already in use?

- How many of you have attended an LML workshop or enough sessions to have a good understanding of what the Loader does and what is required to run the Loader?
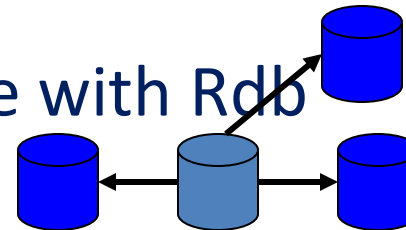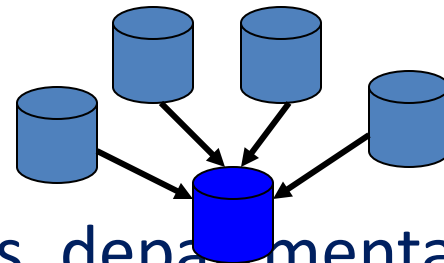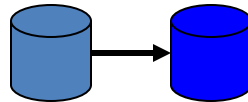
What is the
JCC LogMiner Loader?

- Data changes made in a source database can be published by the Loader to one or more targets.

- The Loader works in near real time or, at your option, in batch-like modes.

- The Loader can support replication, rollup of data from multiple databases (with suitable restrictions on keys), auditing of the source, and other options built on flexible definition of subsets of the source tables and/or columns, filters, and addition of materialized information.

- The Loader monitoring tools are so useful that they provide a window into the overall architecture.

Prime Uses

- Replication
  - Complete
  - Partial
- Combine "regional" databases
- Separate segments for web access, departmental use, or performance
- Inclusion of tools not available with Rdb
- Conversion
- Archival and audit
- Capture of real world scenarios for regression testing and tuning

Requests from Customers

- Past requests and responses
  - Added targets
    - Oracle, Tuxedo, XML (to an API), JDBC (to reach SQL server, and others)
  - Tuned issues with specific targets
    - For example, Oracle handling of dates, trim, null, and others
  - Solved other issues
    - For example, it is *possible* for LogMiner to block back up without the Loader heartbeat.

- Now →   Oracle as a source

Why do the current users want this expansion?

- Mixed shops
  - Some applications were developed against an Rdb database; some were developed against an Oracle database.
  - Reasons for the mix may include changing needs, perceptions, politics, or merger of two enterprises.
  - A way to "roll up" the data in both the Rdb and Oracle databases is often needed.

- Conversion
  - The current Loader can populate a database that is the target of a conversion from Rdb. Further, with the Loader, this is not a one time conversion, but a continuous process as some applications continue to update the Rdb source.
  - The current Loader can also restore the target if development efforts aren't perfect.
  - The current Loader hasn't been able to provide a safety net of **publishing back to Rdb** changes made to the conversion target.
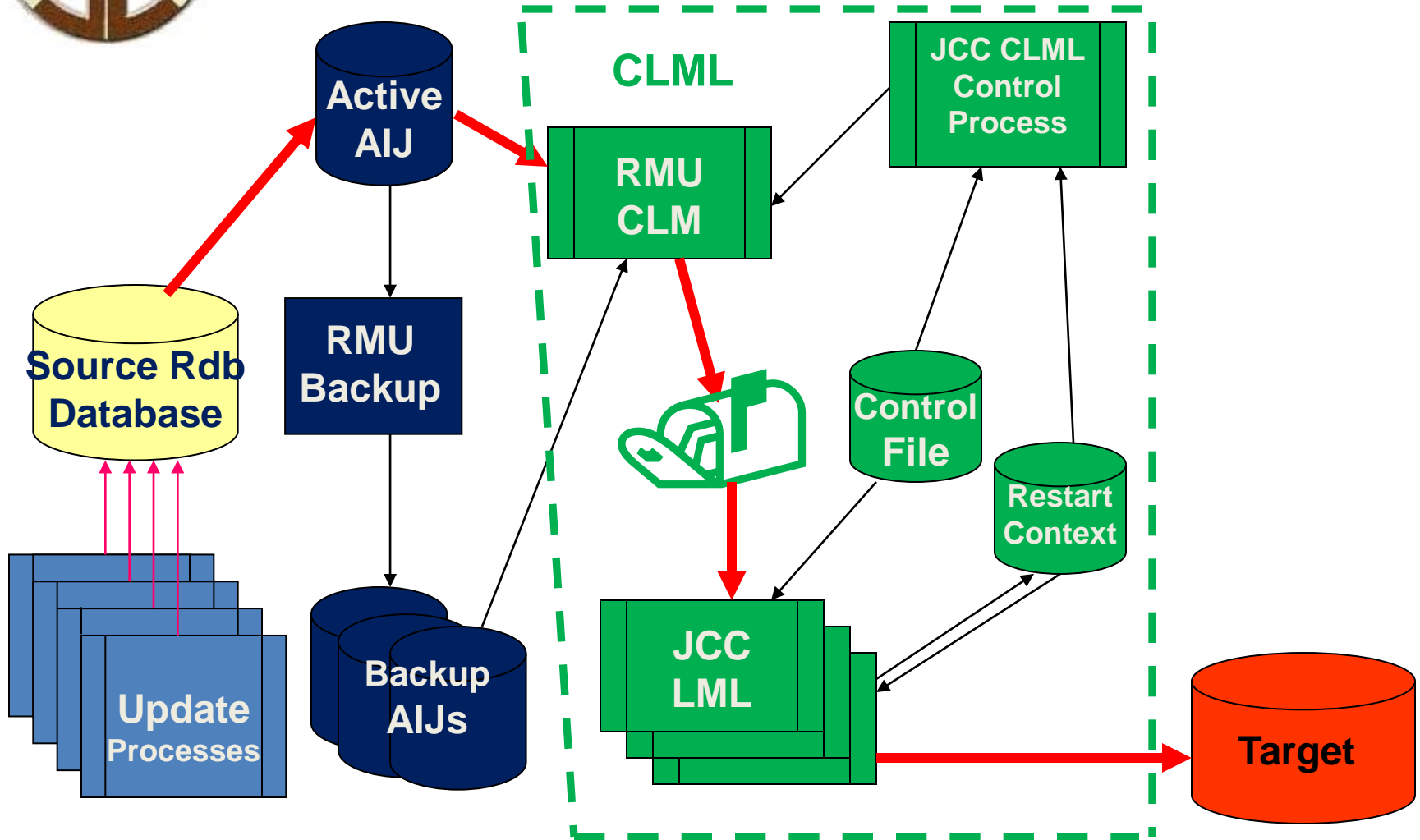
Rdb Source

- The Rdb product includes a tool called the Oracle Rdb LogMiner.
  - The JCC LogMiner Loader and the Rdb LogMiner were developed together.
  - The LogMiner captures data changes from the journals and passes them to a mailbox where the Loader collects them.

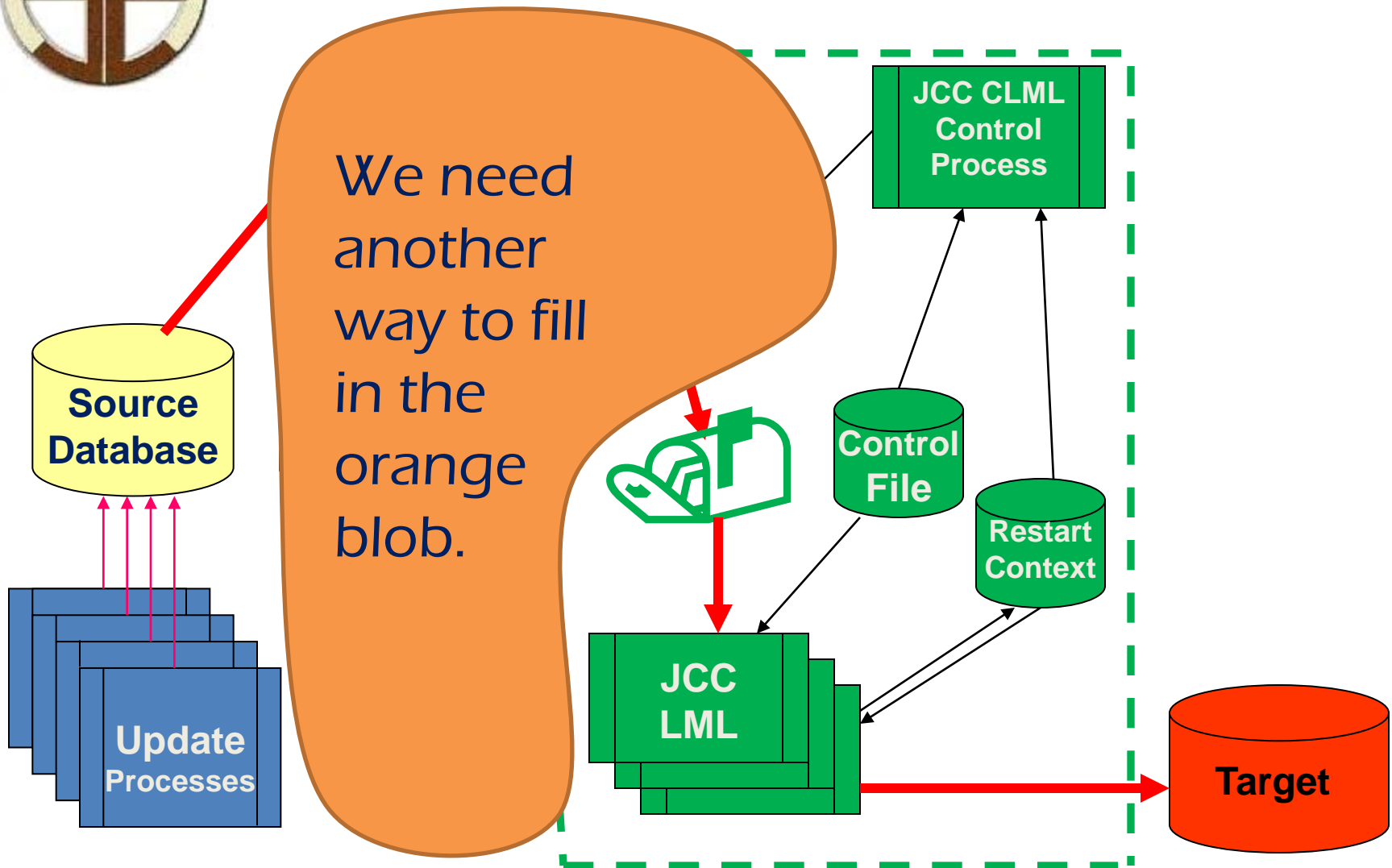# Rdb Continuous LogMiner & the Loader

Oracle LogMiner

- The Oracle 9, 10, 11 products include a tool *called* LogMiner.

- The Oracle LogMiner
  - Is a journaling tool and is used by DBAs to fix data.
  - Has many capabilities, but only a few are exposed for use.
  - Requires polling by the process using the LogMiner.
  - Originally did not work at the speeds required.

- With later versions, the Oracle LogMiner
  - Is one of the components supporting Oracle streams.
  - Is useful in JCC LML for Oracle, but not in the same way as Rdb's LogMiner.

Without the Rdb Continuous LogMiner …

We need another way to fill in the orange blob.

Source Database

Update Processes

JCC CLML Control Process

Control File

Restart Context

JCC LML

Target

Requirements on Change Capture

- Operate on active database log files
- For restart, operate on archive logs & switch to active logs
- Low impact on Mission Critical systems
- Restartable, no data loss
- Fast, minimize latency between commit time and extraction time
- Tunable

Plus All the Features of the Loader

- Extensive monitoring and aids to performance analysis
- Multiple types of targets
- Multiple concurrent targets
- Configuration options
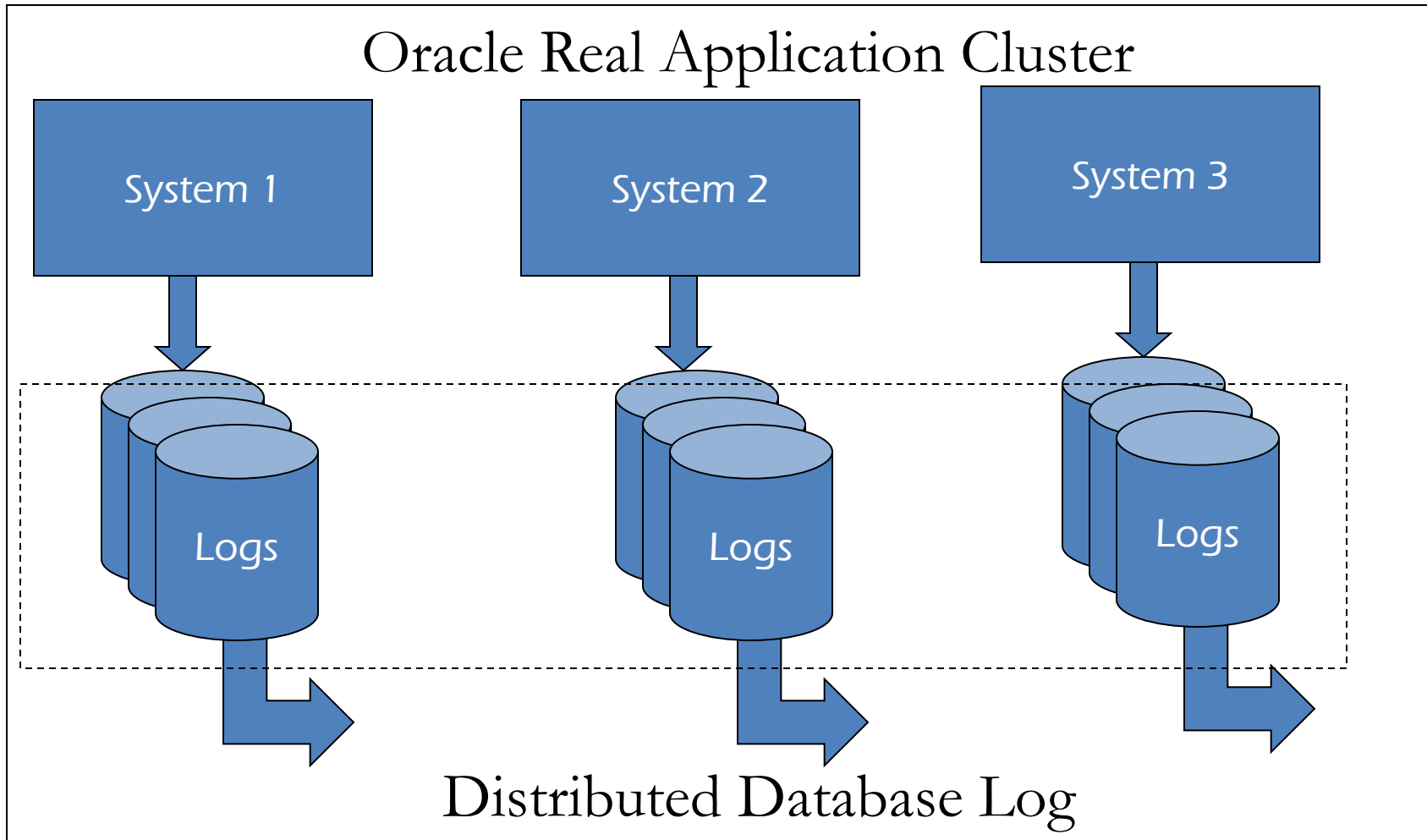- Filters

Requirements for Oracle Logs

- The Oracle database must be in archivelog mode.

- The transactions must avoid "nologging".

- The logs must be managed in such a way that the restart information is not removed while it is still needed.
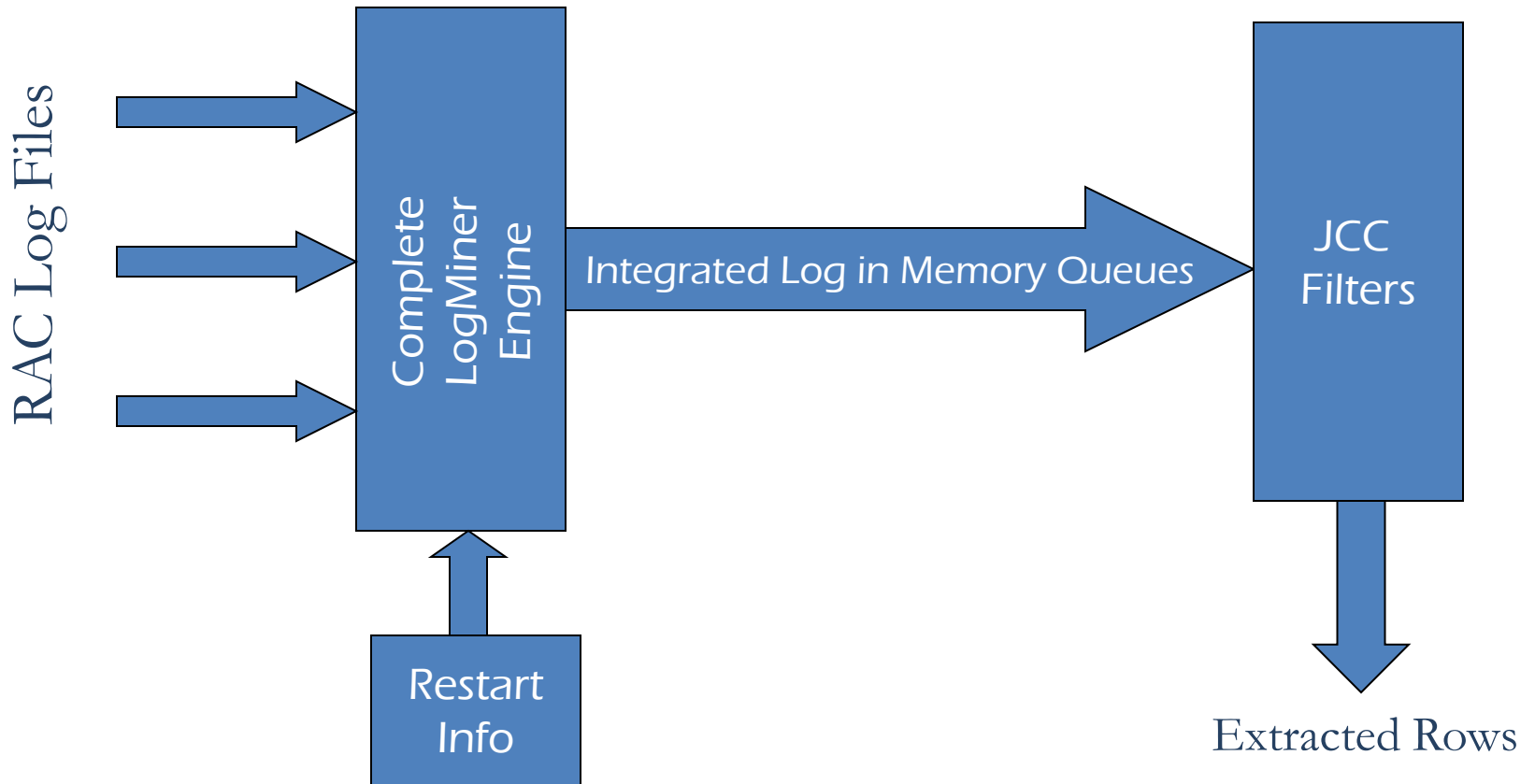
Additional Requirements

- To use the Loader with the changes that are captured,
  - We need to capture from Oracle all columns in a row that changes.
  - We need to capture transactions in the order in which they commit.
  - We need to synthesize commit rows.
- We need to be able to work with Oracle RAC (Real Application Clusters) as well as simpler cases.

Distributed DB Log in a RAC

Oracle Real Application Cluster

| System 1 | System 2 | System 3 |

Logs   Logs   Logs

Distributed Database Log

Oracle Streams

RAC Log Files

Complete LogMiner Engine

Integrated Log in Memory Queues

JCC Filters

Restart Info

Extracted Rows

Oracle Streams

- JCC LML for Oracle uses streams to match the Rdb model.
  - Places agent code in the path of captured data changes.
  - Interprets results with access to Oracle's data dictionary.
  - Synthesizes the commit records from the information that we have.
  - Uses an agent on the source machine and an agent on the Loader machine.
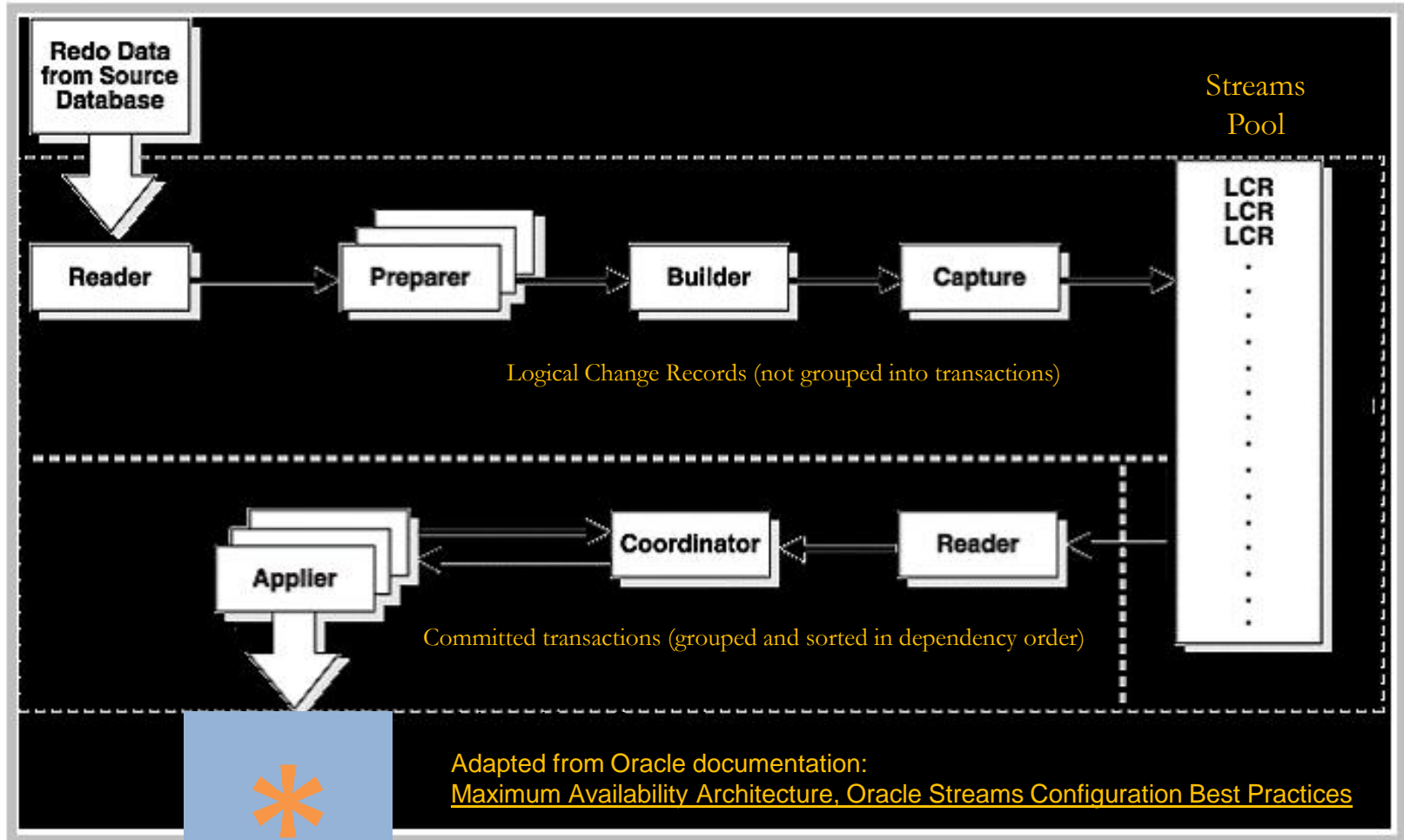
Oracle Streams

- Oracle streams can merge transactions across multiple logs.
- Oracle streams may be restarted as required.
- Oracle streams queues data in memory to filter agents.
- Capture Component
  - Mines redo logs
  - Captures changes
  - Formats a Logical Change Record (LCR)
  - Filters and enqueues
- Apply Component
  - Dequeues LCRs
  - Assembles transactions and calculates transaction dependencies
  - Assigns transactions to available processes

Oracle Streams Processing

Redo Data from Source Database → Reader → Preparer → Builder → Capture

Logical Change Records (not grouped into transactions)

Streams Pool

LCR
LCR
LCR
.
.

Applier ← Coordinator ← Reader

Committed transactions (grouped and sorted in dependency order)

Adapted from Oracle documentation:
Maximum Availability Architecture, Oracle Streams Configuration Best Practices

6/2/2010

20

What is in the Log Files?

- If a row is updated multiple times in a transaction, there will be multiple change LCRs generated for that row.

- With full logging enabled
  - For inserts, all columns with the new data
  - For deletes, all columns with the old data
  - For updates, all columns with the original data and only the changed rows with the new data

Streams and Oracle Versions

- Introduced in Oracle 9i release 2
  - Only accesses archive logs
  - Cannot be real time

- Extended in Oracle 10g release 1
  - Adds support for accessing data in live logs
  - Provides opportunity to get closer to real time

- In 10gR2
  - Added performance enhancements required for JCC LML

- Conclusion: Oracle source must be 10gR2 or later.

Throughput of Streams

- No real, quotable benchmarks were available.
- We have only just begun to test performance.

Loader Challenges

- Data must be moved from the Oracle context into the Loader context.

- The Loader must protect the data
  - Queueing
  - Memory of data in flight
  - Memory of what is processed
  - Security (implemented via encryption of data passed between systems with encrypted pipes)

- Conversion to structures recognized by the Loader
  - To take advantage of the powerful options in the current Loader
  - Formats for
    - Commit rows
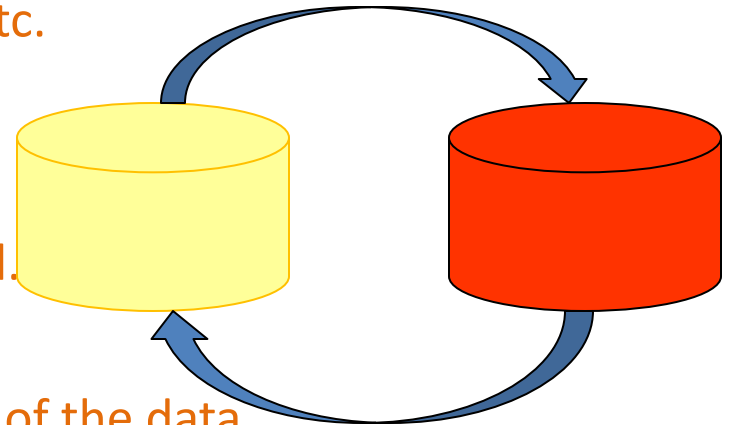   - Data rows

6/2/2010

24

Application Architectures

- For the new product, all of the same application architectures that apply for the original Loader
  - Rollup of multiple databases
  - Replication of subsets to different departments or uses
  - Many more …
- Architectures for both versions of the Loader
  - Mixed shops where Oracle and Rdb databases must share some data
  - Conversion

Architectural Limits

- Cyclic updates
  - When updating both Rdb and Oracle databases and using LML on each, you have the *opportunity* to configure the products such that data is updated in one database, that update is published to the other database, the second database is updated and the data change is published *back to* the first database, etc.

  - Some architectural solution is required.

- Example solutions
  - "Ownership" of all or specific portions of the data.
  - Update flags.
  - Filtering in streams based on "redo tags"

Early Release Limits

- Sources
  - Only Oracle 10gR2 or later with streams and XMLDB *

- Data types
  - Only Rdb data types supported

- Operating systems for the source
  - OpenVMS
  - (We may add HP-UX for the test site client.)

- Operating system for running the Loader
  - VMS

- Row size
  - Individual rows of data of up to 64kb

- Targets
  - All targets supported by the Loader

* This restriction remains.

Expansions & Early Limits

- Expansion beyond the early limits is coming … and coming quickly.

- Time to test the combinations of sources, targets, features, and options, *thoroughly*, is the main reason for these limits.

Additional Limits

- These features will require additional time due to Oracle and Rdb architectural differences.
  - Data Pump
  - Materialized columns
  - Use of the "redo tags"
- Which expansions come first will be influenced by what our customers need.

Inherited Advantages

- All the targets
  - Rdb
  - Oracle 9, 10, …
  - XML to your API
  - Tuxedo
  - JDBC to reach SQL Server and over 100 others
- All the monitoring
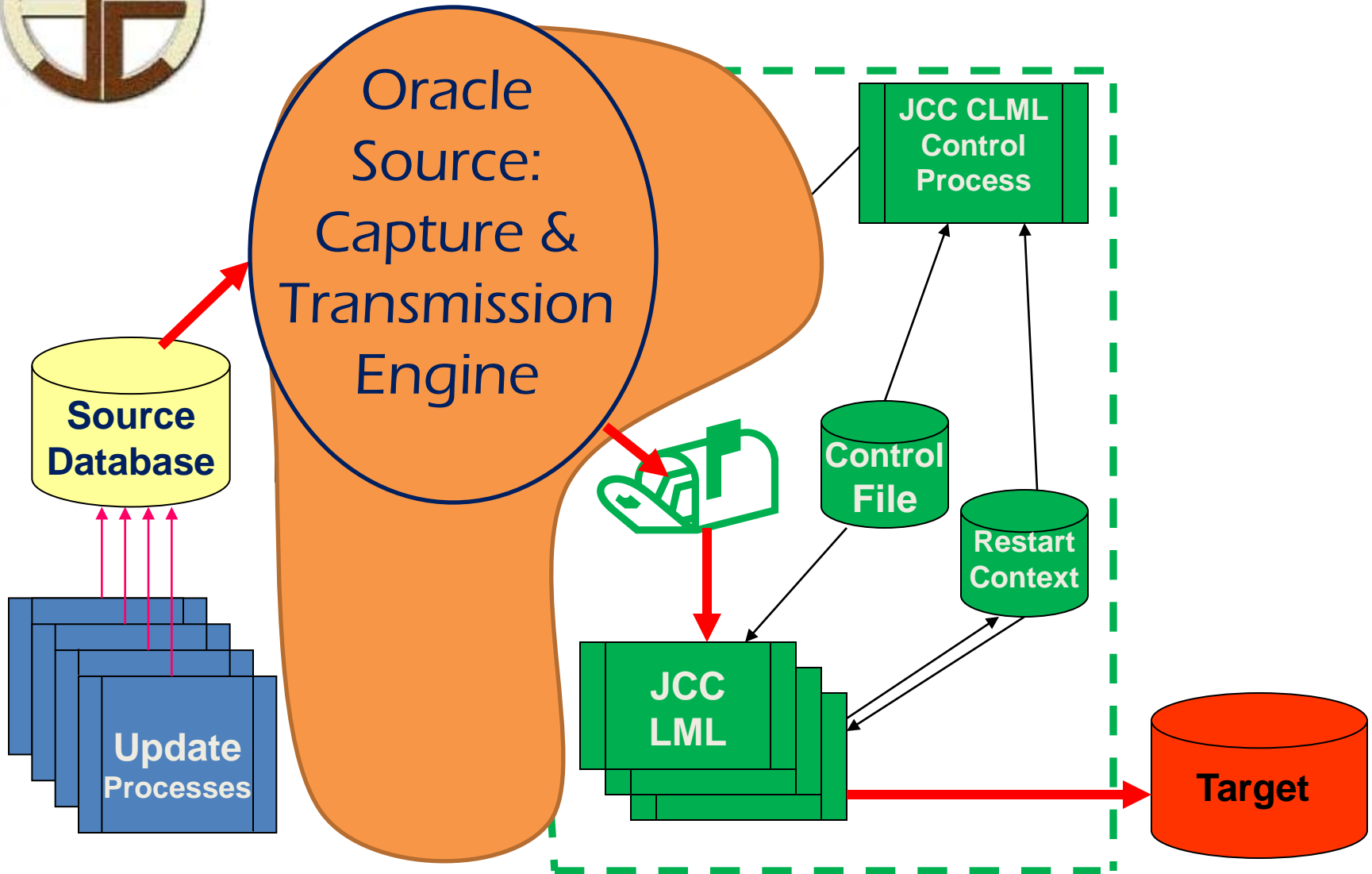- Logging control
- Configuration options

6/2/2010

JCC's LML for Oracle Sources

- We will need to help you understand which options meet your architectural needs.
- We will need to help you establish policies and procedures to
  - Configure
  - Run
  - Stop/pause/manage
  - Monitor
  - Tune
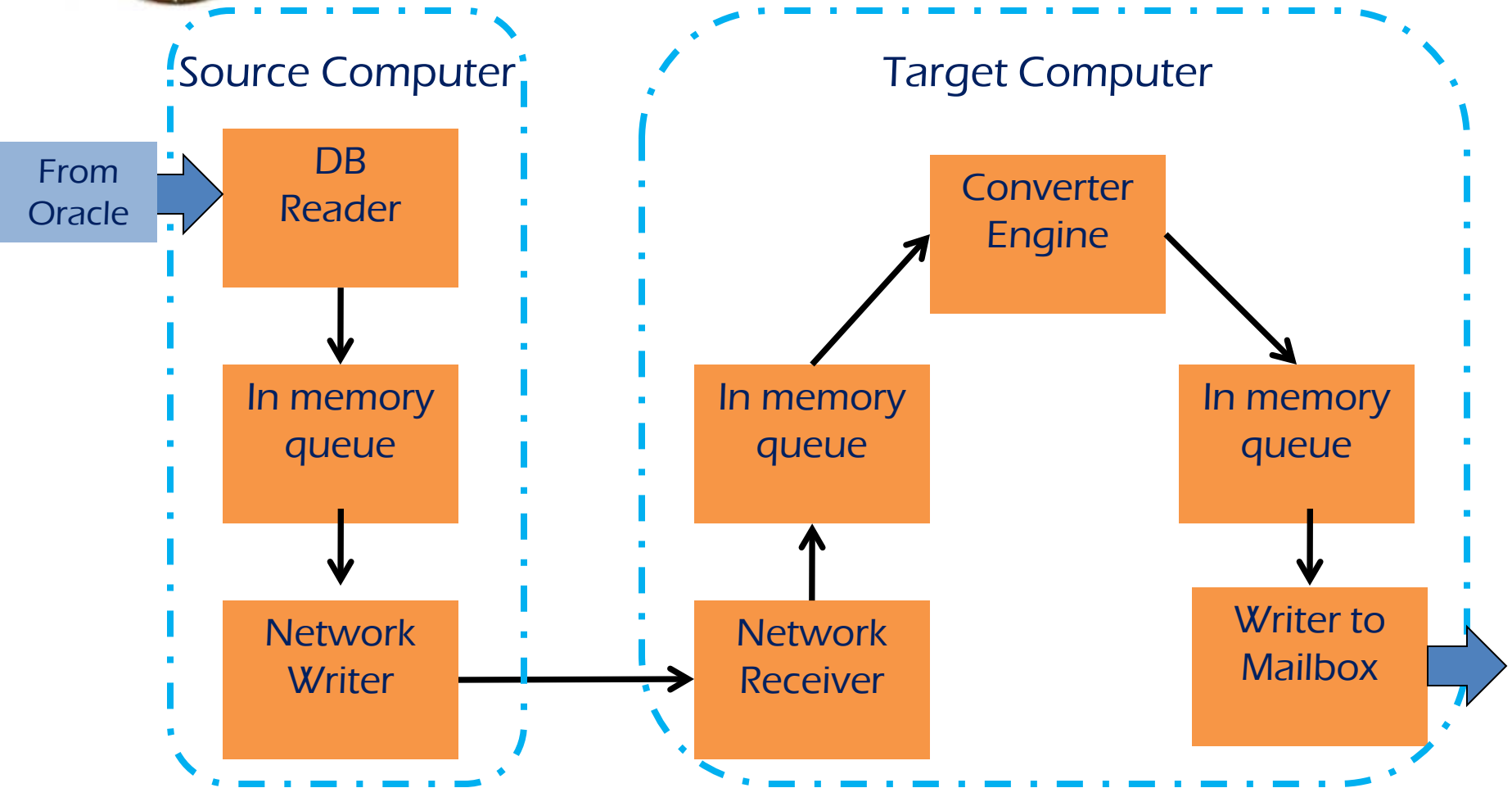- We have the support team to meet this need.

Data Flow with the New Product

Capture & Transmission

Source Computer

From Oracle → DB Reader → In memory queue → Network Writer

Target Computer

Network Receiver → In memory queue → Converter Engine → In memory queue → Writer to Mailbox →

Early Testing

- The new product is currently in testing at JCC.

- In early April – when I submitted the presentation – we expected to go to field test in May.

- We are currently inviting additional field test sites.

JCC Testing

- JCC testing includes
  - Realistic data volumes
  - Random selections of options for the Loader
  - Random "failures"
  - All of the different targets and many other configuration choices
  - Automation that runs the testing until we have tested far more examples than could otherwise occur in the same time
- We still want to test at client sites for
  - Ease of use
  - Realistic transaction rates

Schedules and Options

- Additional field test sites
  - Temporary trial licenses are available at no charge.
  - Basic support is available with trial licenses.
- Production release schedule
  - Help us understand your needs and your schedule.
  - Both Basic and GOLD support will be available.
    - Basic support includes new releases and help during JCC business hours.
    - GOLD support is 24 X 7 X 365.
    - Support references are available.

More Information

- Find descriptions of the LogMiner Loader and other information at http://www.jcc.com/LML.htm

- Slides from numerous presentations from JCC and our clients are available from the Oracle forum sites and at FTP.JCC.COM

- We are happy to answer additional questions.
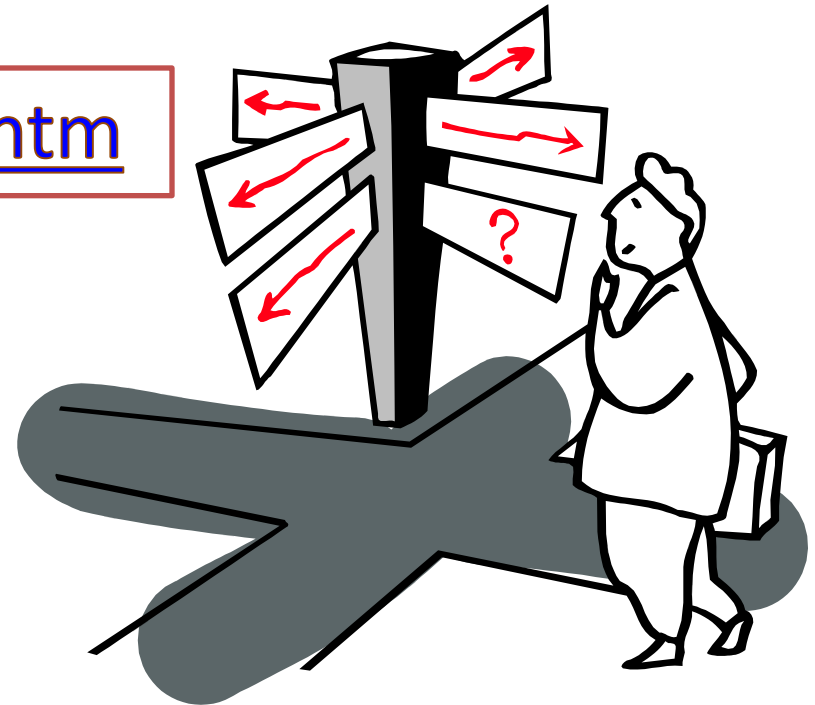 - Send mail to info@jcc.com

Acknowledgements

- Thanks to Oracle engineering for their support and counsel.

- Thanks to our Customers for sharing their experiences with the current Loader.

Questions?

http://www.jcc.com/LML.htm

For more information send mail to info@jcc.com

Join the Conversation

Join the worldwide Rdb community. Send mail to

OracleRdb-request@JCC.com with "SUBSCRIBE" in the body of the message.